Auditable blockchain voting system – the blockchain technology toward the electronic voting process

Aneta Poniszewska-Marańda*,
Michał Pawlak and Jakub Guziur

Institute of Information Technology,
Lodz University of Technology,
Łódź, Poland
Email: aneta.poniszewska-maranda@p.lodz.pl
Email: michal.pawlak@edu.p.lodz.pl
Email: jakub.guziur@edu.p.lodz.pl
*Corresponding author

Abstract: There exist many different electronic voting solutions and each has its own set of advantages and disadvantages. Most of the existing systems suffer from inadequate transparency and auditability. However, recently introduced blockchain technology may provide a solution to these problems. In this paper, auditable blockchain voting system (ABVS) is presented. It combines existing voting approach and combines it with blockchain technology to create a supervised and remote internet voting system, which is transparent and audit capable. The paper describes the system's processes, components, model and results of initial testing.

Keywords: e-voting; electronic voting; blockchain; audit; verification; e-voting system.

Reference to this paper should be made as follows: Poniszewska-Marańda, A., Pawlak, M. and Guziur, J. (2020) 'Auditable blockchain voting system – the blockchain technology toward the electronic voting process', *Int. J. Web and Grid Services*, Vol. 16, No. 1, pp.1–21.

Biographical notes: Aneta Poniszewska-Marańda works continuously since 1998 in the Institute of Information Technology at Lodz University of Technology, currently as an Associate Professor. Her research interests include software engineering, information systems security, analysis and design of information systems, multi-agent-based systems, cloud computing and internet of things. She has published more than 100 research papers in journals, conference proceedings and books.

Michał Pawlak is a PhD student at the Faculty of Technical Physics, Mathematics and Information Technology, Lodz University of Technology. His research interest lies in a field of software engineering. More specifically, his work focuses on blockchain technology and its possible application in different administration, public and private sector solutions.

Jakub Guziur is a PhD student at the Faculty of Technical Physics, Mathematics and Information Technology, Lodz University of Technology. His research interest lies in a field of software engineering. More specifically, his work focuses on blockchain technology and its possible application in different administration, public and private sector solutions.

This paper is a revised and expanded version of a paper entitled 'Voting process with blockchain technology: auditable blockchain voting system' presented at 10th International Conference on Intelligent Networking and Collaborative Systems, INCoS-2018, Bratislava, Slovakia, 5–7 September 2018.

1 Introduction

A new digital currency called Bitcoin was introduced in 2008 by an individual (or a group) under pseudonym Nakamoto (2008). Blockchain technology, upon which the new currency was based on, became considered to be potentially revolutionary in many fields, for example, financial and governance (Zhao et al., 2016). Blockchain technology, in simple terms, is a system of ledgers contained in a chain-like data-structure of interconnected blocks that is stored in a peer-to-peer network of nodes, which collectively validate and negotiate the contents of the whole chain via a dedicated algorithm (Drescher, 2017; Morabito, 2017). Due to its many advantages and possible applications, the technology is receiving a lot of attention and its potential usages are thoroughly researched (Zhao et al., 2016; Risius and Spohrer, 2017). It was discovered that one of the possible applications lies in a field of electronic voting (e-voting).

A democracy would not exist without voting and for this reason a voting process is protected by complex security measures. Despite them all, it is not free from frauds and manipulations (De Faveri et al., 2016; Lehoucq, 2003). Most modern voting systems are generally slow and manipulation-prone. This is a result of their dependency on ballots, which have to be collected and counted by a single central institution. Obtained results cannot be verified by voters, which have to trust the presented results. Moreover, the voters do not have any way to ensure that their votes were handled fairy and correctly.

To solve these problems, e-voting systems were created (Willemson, 2018). The systems used today are far from perfect and have different issues with authentication, privacy, data integrity and transparency (De Faveri et al., 2016). Blockchain technology may provide a solution to the mentioned problems of e-voting systems. The technology allows creation of platforms for public validation and verification of data stored within a chain. This would enable voters to audit voting results without dedicated institutions and their officials. Research and implementation of blockchain-based e-voting systems is ongoing in some countries (Ojo and Adebayo, 2017; Enterprise Estonia, 2012). Most notably, in 2017, South Korea conducted a successful community voting and Sierra Leone conducted a nationwide election using Agora blockchain system in 2018 (Akwei, 2018).

Currently available blockchain-based e-voting systems have many advantages. The most significant one is ability to securely and anonymously cast votes via connection to the internet. However, these systems still face issues with identification and authentication. In most cases, these processes are left to various election officials or depend only on cryptography, which removes all benefits of remote voting and creates a possibility of voter impersonation respectively.

The goal of this paper is to describe a full end-to-end verifiable e-voting system based on blockchain technology. The system is intended to be an enhancement of the existing voting procedure in Poland. The main goal of the system is to provide voters with the

ability to verify the results of the voting and possible ability to follow their own votes to check their correctness.

The remaining parts of this paper are divided into five sections. In Section 2, the theoretical and technical aspects of blockchain technology and e-voting are described. In Section 3, an overview of works related to this field is presented. In Section 4, the original blockchain-based e-voting system is described.

2 Background

This section presents backgrounds of blockchain technology and a theory behind e-voting. Each of these topics is described in a dedicated subsection.

2.1 Blockchain technology

Blockchain technology is a combination of two main elements (Drescher, 2017; Xu et al., 2017):

1 blockchain data-structure

2 blockchain system or network.

Data in blockchain is stored in a chronologically ordered list of interconnected units called blocks, which form *blockchain data-structure*. These blocks are made of a block header and transaction data. The block header is made of three main elements:

1 block metadata, which stores information about a block itself (e.g., index and creation timestamp

2 hash reference to a previous block, which is created by application of a hash function to contents of the previous block

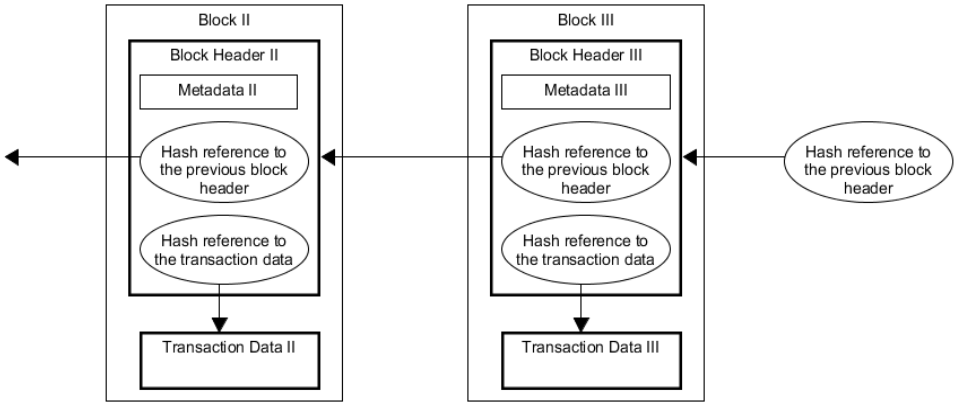
3 hash reference to stored transaction data, which are generated by application of a hash function to the transaction data.

The transaction data is a list of transactions and their respective data saved within the block. Figure 1 presents an exemplary model of a blockchain data-structure and how its elements are connected to each other. Main components are represented by rectangles with thicker lines. Standard rectangles represent metadata ovals designate hash references and arrows illustrate connections made by the references.

As mentioned previously, the hash references are generated by application of one of different cryptographic one-way hash functions to contents of a block or transactions in each block. A hash function maps data of an arbitrary size to a unique bit string of a fixed size called hash value or hash reference. Due to the functions' properties, hash values are easy to calculate but very difficult to invert from the point of view of computational theory. Furthermore, hash functions are input-sensitive, which means that any change of a provided input will result in a change of a resulting hash value (Stallings, 2013). This property ensures immutability of a blockchain because any change of contents of any block would force a modification of hash references in all subsequent blocks. Finally, it is important to note that generated hash values are unique, which allows identification and tracking of each block. This in turn allows verification of their correctness. Different

implementation of blockchains utilise different hash functions, the most popular functions used include SHA256, RIPEMD160, Merkle trees and the elliptic curve digital signature algorithm (Morabito, 2017; Karame and Audroulaki, 2016).

Figure 1 Structure of a blockchain data-structure



Blockchain system or network is a distributed peer-to-peer network of connected nodes, which store and negotiate information contents of a blockchain. Nodes receiving new transactions propagate them to other known nodes until a whole network is aware of the new transactions. Each node store their own copy of the blockchain and add new blocks to in a process called *mining*, which consists of validation and aggregation of stored transactions into blocks which are appended to the chain. New blocks are broadcasted to the network in order to update other chains. To ensure consistency of the stored chains, the system strives for a consensus about which new blocks can be added. The consensus is reached through *consensus algorithms*, which have many different forms. The most common include (Xu et al., 2017; Mingxiao et al., 2017; Zheng et al., 2017; Castro and Liskov, 1999):

1 proof-of-work

2 proof-of-stake

3 delegated-proof-of-stake

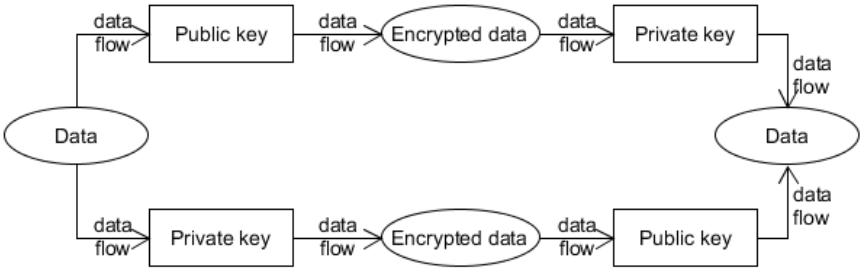
4 practical-byzantine-fault-tolerance.

Proof-of-work algorithm makes all nodes compete with each other in solving a computationally expensive mathematical problem. The first one to solve the puzzle can add a new block to a chain and receives a reward. The new block is broadcasted to other nodes for peer validation. Proof-of-work algorithm assumes that the longest chain is authoritative one due to the greatest amount of computational total work. *Proof-of-stake* is based on ownership of a digital currency with assumption that an owner of a large amount of a currency would not have any incentive to tamper with or attack the network. To prevent centralisation of such network various methods of authoritative node selection have been proposed, for instance, random selection or ‘age and size’ of a coin set. *Delegated-proof-of-stake* is a variation of the proof-of-stake algorithm in which

authoritative nodes are selected by other nodes. *Practical-byzantine-fault-tolerance* selects an authoritative node in a three phase round, in which phases are advanced when a node receives votes of more than 2/3 of all nodes in a network.

Blockchain technology utilises asymmetric cryptography for authentication and authorisation. This approach uses public and private key pair, which can be used for encryption and decryption of a message. A message encrypted with one key can only be decrypted with the other (Figure 2).

Figure 2 Idea behind asymmetric cryptography



There are two ways in which the key pair can be used. Public-to-private method encrypts messages with a public key available to everyone and decrypts it with a corresponding private key. This usage corresponds to a mailbox, which can receive messages from anyone, but only its owner can open it. Blockchain systems use this method to secure information exchange. Private-to-public method encrypts messages with a private key and decrypts them with a public key. This is a method of proving ownership due to the fact that only the owner of the private key could create a message, which can be decrypted with the corresponding private key. Blockchain systems use this method for transaction authorisation (Morabito, 2017; Karame and Audroulaki, 2016).

Combination of all these technologies and solutions gives blockchain-based systems many advantages over traditional storage systems. Blockchain-based systems are censorship-proof due to a lack of a central authority in their networks. From the point of view of data integrity, they are very secure on transaction and system levels. Blockchain provides transparency as transactions are conducted and verified publicly and contained in a blockchain data-structure stored on every node of a system, which makes it available to anyone. Another potential advantage is possibility of obtaining data almost in real time (depending on a given system). Due to these characteristics, blockchain technology may provide a solution to the most common issues of e-voting systems, namely, transparency, auditability and modification resistance.

Systems based on blockchain technology are constantly developed and many different implementations exist. In general, they can be classified by two characteristics (Drescher, 2017):

1 read rights

2 write rights.

Two types of blockchain systems can be distinguished from the point of view of *the read rights*:

1 *public blockchain systems*, which allow all users and nodes to access a system's blockchain and its contents

2 *private blockchain systems*, which allow only a selected group of users and nodes to access a system's blockchain and its contents.

The write rights divide blockchain systems into:

1 *permissionless blockchain systems*, in which all users and nodes can participate in a given consensus algorithm

2 *permissioned blockchain systems*, in which participation in a given consensus algorithm is restricted to a selected group of users and nodes.

As mentioned previously, blockchain-based systems are constantly developed and there exists many implementations of this technology. The best know example is Bitcoin virtual currency, which was its first practical implementation. Bitcoin is a fully distributed, public and permissionless system utilising proof-of-work consensus algorithm. Ethereum Platform created by Ethereum Foundation (2014) is the second most well-known implementation. It is distributed, public and permissionless platform using proof-of-work and smart contracts. The platform provides infrastructure for creation of blockchain-based applications. Lastly, multichain is a platform for private blockchain systems, which utilises consensus algorithm similar to practical-byzantine-fault-tolerance (Coin Sciences, 2015). It was designed for use within organisations and for financial transactions.

2.2 E-voting concepts and systems

E-voting can be defined as any type of election or referendum that utilises electronic means to facilitate voting procedures at minimum for casting votes (Caarls, 2010). The term e-voting covers a variety of different systems, solutions and implementations. Such systems provide many benefits including:

1 fraud prevention through reduction of a human involvement

2 results processing acceleration

3 reduction of a number of spoiled ballots by improving presentation and automatic validation of ballots

4 cost reduction through minimising voting process overhead

5 increase of involvement in democratic processes by increasing availability (e.g., remote voting)

6 potential for more direct democracy (Wolf et al., 2011).

However, implementation of any e-voting system is always connected with numerous technical, procedural and legislative challenges. One of the most substantial one is a lack of trust in e-voting systems. There are many causes of this, the most important ones are:

1 inadequate transparency and understanding of such systems by non-expert

2 lack of widely accepted standards, which reduce confidence in systems' dependability

3 vulnerability to attacks and manipulations by privileged insiders and system providers

4 increased costs of voting due to required infrastructure, for example, power supply, communication technology, etc. (Wolf et al., 2011).

E-voting systems follow the same procedure as traditional voting systems. The procedure consists of six phases:

1 voter registration conducted personally or by an authority

2 authentication, to verify voters' identities

3 authentication to ensure only eligible voters can vote

4 vote casting

5 vote counting

6 vote verification in which voting process is checked for possible frauds and errors.

Furthermore, all e-voting systems must fulfil a common set of requirements before they can be considered valid. The requirements include (De Faveri et al., 2016; Schneider et al., 2017; Fouard et al., 2017; Zhou et al., 2016):

1 voter identification and authentication

2 voter privacy assurance

3 correctness

4 transparency

5 verifiability

6 ballot integrity

7 availability

8 fairness.

Voter authentication and authorisation describes a requirement of ensuring that only eligible citizens are able to cast their votes. *Voter privacy assurance* property refers to a need of ensuring only voters themselves know values of their votes and they cannot be connected with their votes. *Correctness* enforces that each voter has only a single vote and that only valid votes are counted, while invalid ones are disregarded. *Transparency* requirement refers to a need of e-voting systems to be open to verification and understandable for non-experts. *Verifiability* requires e-voting systems to allow independent third party to check their correctness. *Ballot integrity* ensures that votes are immutable after being cast. *Availability* requires e-voting systems to be accessible by anyone in an election time-frame. *Fairness* requirement ensures that all participants have equal chances and does not have any advantages form systems themselves.

E-voting systems can be differentiated according to many different criteria. The most general way of classification is related to two characteristics (National Democratic Institute, https://www.ndi.org/e-voting-guide/common-electronic-voting-and-counting-technologies):

1 remoteness

2 supervision.

In general, e-voting systems can be *remote*, which means that cast ballots are transmitted through a communication channel to a central location for aggregation and counting, or *non-remote*, in which cast ballots are collected locally before counting. On the other hand, *supervised* systems allow voting only from a location controlled by some voting authority, while *unsupervised* systems allow voting from any location without outside control.

In addition, e-voting can be classified into four types in respect to their use of information and communication technologies in a voting process. These four types consists of:

1 voting with dedicated voting machines

2 voting with optical scanning voting machines

3 voting with electronic ballot printers

4 voting through the internet (Wolf et al., 2011; National Democratic Institute, https://www.ndi.org/e-voting-guide/common-electronic-voting-and-counting-technologies; United Nations Development Program, 2016).

Voting with dedicated voting machines utilises electronic devices, which through keyboards, touch screens and other input devices record votes. Depending on the system, devices can transmit votes or store them internally. Often, these machines are also accompanied by *voter-verified audit paper trails* (VVAPTs) that are printed copies of the recorded votes used for verification procedures. Voting with dedicated voting machines provide relatively fast data collection and vote counting. It prevents ballot spoiling due to better ballot presentation. On the other hand, this approach is expensive due to costs of deployment and maintenance. Dedicated voting machines are most often created by third parties, which makes verification of results impossible because software and hardware cannot be inspected on every single device (Wolf et al., 2011; National Democratic Institute, https://www.ndi.org/e-voting-guide/common-electronic-voting-and-counting-technologies; United Nations Development Program, 2016).

Voting with optical scanning machines, the voting consists of scanning a readable paper ballot with a dedicated machine. This system is easily understood by common voters because this does not change traditional voting process. Like the previous approach, it provides fast and accurate results. However, it depends on paper ballots, which are not tamper-proof and it is also expensive to deploy and maintain. Furthermore, the devices suffer from the same transparency problems as the previous group (Wolf et al., 2011; National Democratic Institute, https://www.ndi.org/e-voting-guide/common-electronic-voting-and-counting-technologies; United Nations Development Program, 2016).

Voting with electronic ballot printers is similar to dedicated voting machines. However, they produce readable paper recipes or tokens, which can be disposed in ballot boxes or read by some counting machine instead of recording votes themselves. This approach is transparent and easy to verify due to remaining physical trail of all votes. On the other hand, it is expensive due to deployment and maintenance requirements. Furthermore, its only advantage over traditional voting systems is prevention of

ballot spoiling (Wolf et al., 2011; National Democratic Institute, https://www.ndi.org/e-voting-guide/common-electronic-voting-and-counting-technologies; United Nations Development Program, 2016).

Finally, *voting through the internet* consists of voting through devices connected to the internet, which transmits votes to a designated location for verification and counting. It can take a form of a supervised and non-remote voting as well as unsupervised and remote voting. This last form is potentially the most advantageous and most desirable form of voting. Voting through the internet does not require dedicated machines and can provide fast and accurate results almost in real time. Unfortunately, this type of voting has the most numerous security issues, which include:

1 hacker attacks

2 potential lack of anonymity and privacy

3 'creation' of votes or unauthorised votes

4 influencing voters by third parties

5 vulnerability of data during transmission (Wolf et al., 2011; National Democratic Institute, https://www.ndi.org/e-voting-guide/common-electronic-voting-and-counting-technologies; United Nations Development Program, 2016).

Additionally, e-voting systems can be further differentiated by utilised cryptographic primitives and schemes. The most common primitives used in e-voting include:

1 *Zero knowledge proofs* that enables different parties to prove one another that they know some information without revealing it.

2 *Secret sharing* that distributes secrets among groups in parts, so no single participant has the whole secret.

3 *Homorfic encryption* than enables parties to conduct valid operation on encrypted data without decrypting it.

4 *Blind signatures*, which allows signing encrypted data without decrypting it.

5 *Mix-net* that allow creation of difficult to track communication by creation of a network of nodes, which shuffle every message before propagating them further (Fouard et al., 2017; Zhou et al., 2016).

E-voting systems can be analysed form different points of view and as can be seen each type of e-voting system has different advantages and disadvantages. Some of them are directly tied to the e-voting concept itself and some come from a specific approach or implementation.

3 Related works on e-voting systems

Numerous publications about the topic of e-voting have been published. In Fouard et al. (2017), the authors present an overview of existing e-voting schemas. The paper reviews properties and the most popular cryptographic primitives of e-voting schemas. In addition, the paper presents 16 e-voting schemes and compares them.

Schneider et al. (2017) present the current state of the e-voting field. Similarly to the previous work, the paper describes different schemas classified according to used cryptographic primitives. Furthermore, the authors illustrate how e-voting systems can be attacked and discuss existing e-voting systems and issues form which they suffer.

Although there are no widely and officially accepted standards for e-voting systems, there exist a few documents that attempt to create some norms and guideline. The Council of Europe created two of such documents. The first one is Council of Europe – Committee of Ministers (2017), which contains recommendations for conducting elections with usage of electronic means in a form of a checklist. The second one is CAHVE (2017) and it explains the previous document in more detail and presents detailed technical recommendations. In addition, Caarls (2010) provides guidelines for introduction of e-voting system in election procedures.

Another standardisation attempt was made by The International Institute for Democracy and Electoral Assistance (International IDEA). Effects of its work are presented in Wolf et al. (2011) that is composed of guidelines, recommendations and considerations for implementation and introduction of an e-voting system.

In the previous sections, the most important requirements of e-voting systems were presented. In Mello-Stark and Lamagna (2017), transparency and verifiability are closely examined. The authors analyse methods of auditing e-voting systems and provide a brief overview of existing solutions, i.e., Helios and WAVE.

Being a relatively new technology, new applications of blockchain are still being researched. Ojo and Adebayo (2017) describe ongoing research, conducted by the Digital5 (D5) countries, on possible blockchain application in e-governance, including e-voting. The most active countries are Estonia and South Korea. The letter of which conducted a successful community voting in 2017 using a blockchain-based system.

As mentioned previously, there exist many e-voting solutions and implementations. One such system is presented in Ochoa and Peláez (2017). SAVE is a supervised and remote e-voting system for medium and large-scale voting, for instance, university elections. The authors describe components and processes of the system. The system uses commonly available devices as voting machines, for example, smartphones and personal computers. SAVE utilises symmetric encryption for software signing, asymmetric encryption (RSA with 2,048 bit key length) and HMAC-SHA256 for message authentication. In addition, the system produces VVPATs required for verification. They are obtained with simple printers.

The most successful e-voting system today is used in Estonia. It is called i-voting and was introduced in 2005 (State Electoral Office of Estonia, 2017). I-voting is a remote and unsupervised internet voting based on ‘envelope scheme’ (Heiberg and Willemson, 2014). In order to cast votes, voters must complete an authentication process using their ID-cards or mobile phones with special SIM cards with encrypted ID of their owners. The voters can cast multiple votes but only the most recent one is counted. This is done to prevent manipulation of the voters. In addition, a dedicated application is used for verification of the votes and it can be used by anyone. Finally, it is important to note that i-voting is being constantly improved and upgraded.

The main example of existing blockchain-based e-voting systems is Agora Technologies (2015). It is a multi-layer system intended to be customisable. Agora provides a possibility of conducting supervised and unsupervised remote voting. It utilises a hybrid of permissionless and permissioned public blockchain. It was implemented in Sierra Leone, which was able to conduct a successful voting in 2018. In

general, the system leaves authentication and authorisation to election officials but it provides a possibility of facilitating this process using offered system based on digital signatures.

Another blockchain-based system is presented in Bistarelli et al. (2017). The presented solution is based on Bitcoin and is intended to be end-to-end verifiable. Authentication and authorisation is conducted using a protocol called anonymous Kerberos. The presented system represents votes as tokens, which are smallest transferrable amounts of Bitcoins, fee included. It is assumed that voters must register with election officials before they can participate in the voting. The system fulfils most e-voting requirements with a possible exception of voters' privacy. This is due to a possibility of linking voters with their transactions within a blockchain.

Finally, FollowMyVote (https://followmyvote.com) is an Ethereum-based voting platform for remote and unsupervised voting. Elliptic curve cryptography is used for security of transactions and webcams for identification and authorisation, which is conducted by ID scanning. The platform also allows its users to supervise an election process in almost real time. Furthermore, FollowMyVote allows voters to switch their votes during the election.

4 Auditable blockchain voting system

This section presents auditable blockchain voting system (ABVS), which is intended to be a remote and supervised internet voting system based on blockchain technology for storage and verification of votes. The main goal of ABVS is to enhance the existing voting processes in Poland. The system is in development stage and its prototypes are being developed and tested. The following subsections present ABVS process overview, ABVS components and implementation details.

4.1 ABVS process overview

The voting in ABVS is divided into three phases (Pawlak et al., 2018a, 2018b):

1 election setup and preparation

2 voting

3 counting and verification.

In the *election setup and preparation* phase, the first step is to select trusted public and private institutions, which will act as nodes in the ABVS blockchain network. These institutions will provide computing power and storage capabilities to the system. The second step consists of certification and signing of software and hardware by election officials (which may include representative of the selected institutions). The third step is generation of unique vote identification tokens (VITs), which are used for identification and verification of votes in later phases. Lastly, the VITs and ABVS hardware and software are distributes between polling stations.

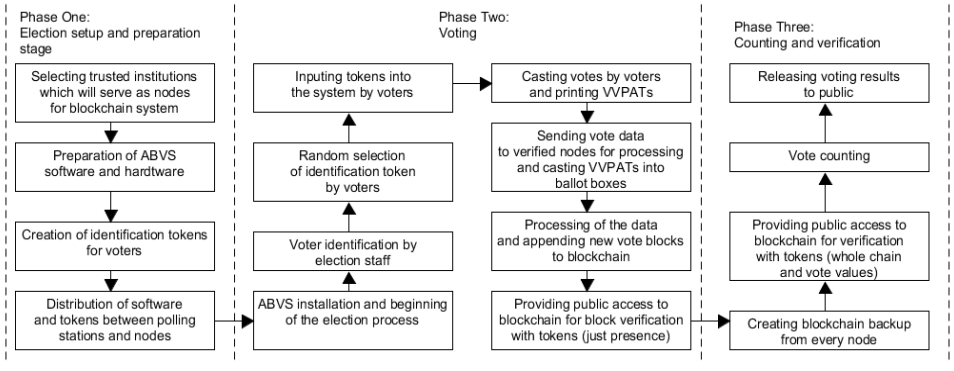
The *voting* phase starts before the actual voting and ends after all election districts are closed. The election officials at polling stations launch the ABVS hardware and

software. The system uses a permissioned blockchain thus ABVS nodes will only accept votes from predefined locations. During the voting itself, voters identify themselves with the officials (presenting ID cards and signing in a registry in case of Poland). Then, the voters randomly select VIT from available at a given station pool. With these steps completed, the voters can cast their votes using available at station voting equipment. Default implementation of ABVS assumes that voting will be conducted on standard personal computers equipped with printers. The voters cast their votes using provided interface and confirm their choices providing their chosen VITs. The votes are broadcasted through a secured and encrypted communication channel to the trusted nodes, which propagate them further and begin verification process using a blockchain algorithm. Valid transactions are added to blocks, which in turn are added to an ABVS blockchain. Meanwhile, the voters receive their VVPATs which they can depose in ballot boxes. Each VVPAT can be mapped to a corresponding transaction in the blockchain via contained within both VIT. The voters then can leave the polling station with their VITs. They can use them to identify, check and verify their vote in the blockchain. At this stage, the whole blockchain can only be viewed at the trusted nodes due to legislative requirements. ABVS does not change the existing voting procedure in the general sense. Due to its supervised nature it also prevents voter impersonation.

The *counting and verification* phase starts after the last polling station is closed, the system is deactivated by the election officials, who then must open and list all unused VITs. The list of the remaining tokens is published for vote verification. Furthermore, backups of the stored blockchains are created at each node and the blockchain is made public, which allows a public verification. With the addition of VVPATS, this enables vote audition and verification. Using VITs, it is possible to compare blocks within the blockchain with VVPATs. In case of any errors, the voters can use a separate dedicated application that allows anonymous error notification. Errors are checked by comparing the value of the given block with the corresponding VVPAT (VVPAT is given priority). Finally, after some designated time elapses, all procedures are closed and final election results are published.

In Figure 3, a summary of the ABVS voting process is presented in a form of a schema. It is split into three phases, which are separated by dashed lines. Rectangular boxes in each phase represent the required steps described in the previous paragraphs. The rectangles are connected by arrows, which denote order of step execution.

Figure 3 ABVS phase overview



4.2 ABVS components

ABVS is made of six components:

1 client applications (at polling stations)

2 network of trusted nodes

3 VITs

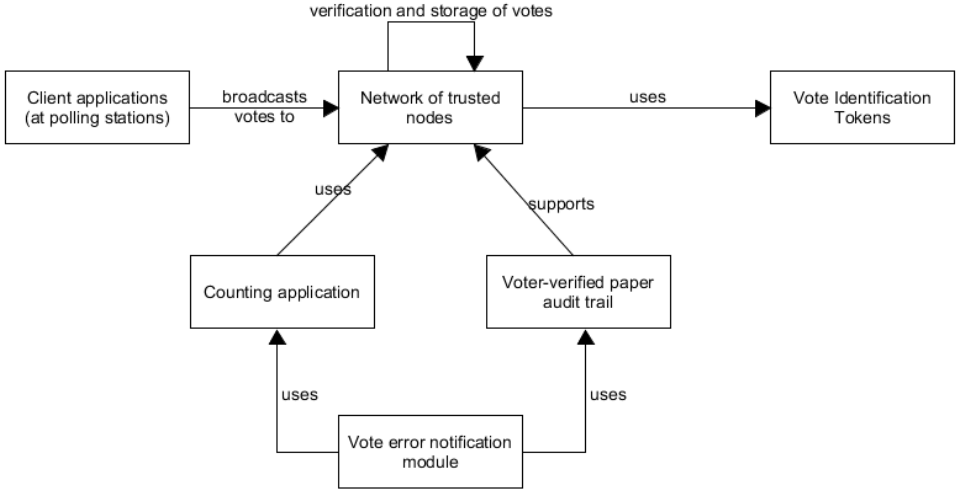
4 voter-verified paper audit trail

5 vote error notification module

6 counting application.

In Figure 4, a model of relations between the ABVS components is presented. The components are represented by rectangles connected by labelled arrows illustrating the relations.

Figure 4 Model of relations between ABVS components



The first component is made of *client applications* which are certified and signed lightweight programs installed on computers at polling stations. They are used for vote casting that takes form of blockchain transactions. Each transaction contains the following information:

1 transaction creation timestamp

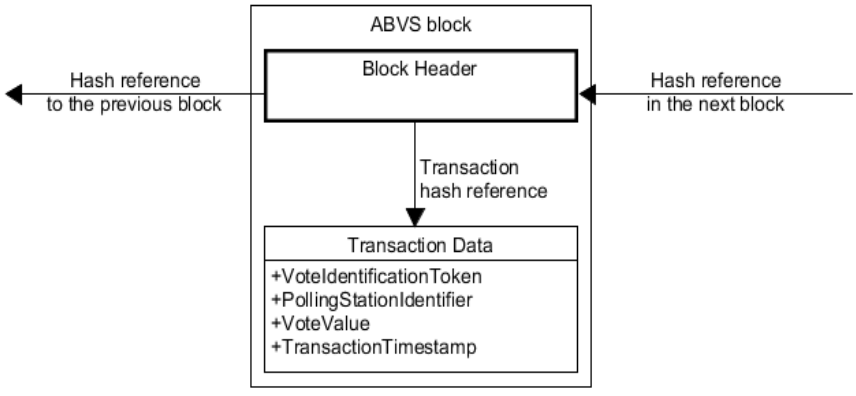
2 vote value

3 VIT

4 polling station identifier.

A model of an ABVS block is presented in Figure 5. When votes are cast, the transactions representing it are broadcasted to an ABVS network of the trusted nodes according to blockchain paradigm.

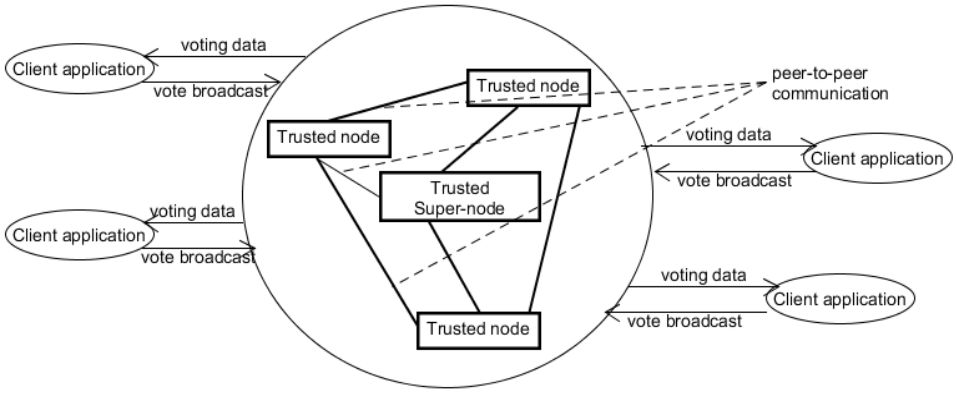
Figure 5 ABVS block model



The second component consists of a *network of trusted nodes* which represent trusted and verified public and private institutions that volunteered to participate in an ABVS voting procedure. The node created by central national electoral authorities (National Electoral Commission in case of Poland) is considered to be a super-node. It is responsible for creating an initial block (*genesis block*) containing election data, namely, election goal, start and ending date, candidates and list of the trusted nodes and a list of all available VITs. The initial block is broadcasted to the trusted nodes at the beginning of the election. It is used for transaction and blockchain verification. Furthermore, the client applications use it as a data source for displayed ballots. When the voters cast their voted, the client application broadcasts the votes (transactions) to the known subset of nodes. Each node broadcast the received transactions to the other nodes and proceeds to mine new blocks (it is assumed that a single block represent a single vote).

In Figure 6, a schema of interaction within the ABVS network of nodes is illustrated. Ovals represent the client applications. Rectangles designate the trusted nodes and labelled arrows represent interactions between the network of trusted nodes and the client applications.

Figure 6 Schema of interactions in the ABVS network



The third component is made of *VITs* which are alphanumerical codes designed to facilitate authentication and authorisation of the voters within the system. They enable vote following and identification during and after the election. VITs can be stored on paper sheets hidden in envelopes, scratch cards or any medium which allow a random selection without revealing its contents. The tokens are generated and distributed during the first phase of the ABVS election.

The fourth component is *voter-verified paper audit trail* that is composed of paper representations of cast votes. Each VVPAT contains the same information as the ABVS block (Figure 5). They are printed in standard printers and disposed into ballot boxes after the voters verify their contents. This provides audit and verification capabilities defined as one of the main goals of the system.

The fifth component is a *vote error notification module* that is responsible for handling anonymous error notification sent by the voters. It is composed of a dedicated applications deployed on a government server which requires providing a valid VITs and an error explanation before a notification can be sent. Valid complaints are verified by comparison between an ABVS block and its corresponding block.

The sixth component is a *counting application* that is responsible for providing results of the election. The application is signed and certified before it can be deployed. Each node of the ABVS network is provided with its own instance, in order to provide fault tolerance and to create multiple comparable results for verification purposes.

4.3 Model of ABVS

ABVSs involve three types of actors:

1 trusted super-node

2 trusted nodes

3 client applications.

Trusted super-node is an initial node of the ABVS voting system. It is located on a server owned by a national election office and only election official can have access to it. *Trusted nodes* are blockchain nodes stored on servers provided by trusted and verified public and private institutions participating in a voting process. *Client applications* are lightweight application for providing voters with a means of interaction with the ABVS network. Each actor provides a set of standard operations required for the system to operate correctly.

Main functions available to the trusted super-node consist of:

1 *ABVS.CreateVoting(candidates, start_date, end_date, params) -> V*, which is a procedure that takes a list of candidates, precise dates of start and finish of a given election and creates and returns voting *V*.

2 *ABVS.RegisterTrustedNode(V, node_address)*, which is a procedure that takes a trusted node address and adds it to a set *N* of trusted nodes participating in the vote *V*.

3 *ABVS.VITGen(V, count, params)*, which is a procedure that for the given voting *V* generates a defined number of VITs.

4 *ABVS.AddPollingStation(V, polling_station_address)*, which is a procedure that takes a polling station address and adds it to a set *P* of valid polling station addresses for the given voting *V*.

5 *ABVS.BeginVoting(V)*, which is a procedure that takes the voting *V* and creates a genesis block out of *V*'s data and propagates it to all registered trusted nodes.

The basic operations for the trusted nodes include:

1 *ABVS.PropagateVote(vote, nodes, params)*, which is a procedure for transmitting the incoming votes to a set of known and active trusted nodes denoted as *nodes* (*nodes* $\subseteq N$).

2 *ABVS.ValidateVote(vote) -> True/false*, which is a procedure for inspecting correctness of the incoming votes.

3 *ABVS.MineVoteBlock(vote) -> B*, which is a procedure that take a given vote and mines and return a new block B.

4 *ABVS.PropagateVote(B, nodes, params)*, which is a procedure for transmitting a new block B to a set of known and active trusted nodes denoted as nodes (*nodes* $\subseteq N$).

5 *ABVS.CalculateResult(blockchain) -> R*, which is a procedure that iterates over the whole blockchain and returns results *R*.

It is important to note that all procedures available to the trusted nodes are also available to the trusted super-node. On the other hand, the trusted nodes do not have access to the procedures available to the trusted super-node.

Finally, the basic operations available to the client applications consist of:

1 *ABVS.GetVotingData(known_nodes)*, which is a procedure that sends a request for voting data to the best node from a set of participating nodes denoted as *known_nodes* (*known_nodes* $\subseteq N$).

2 *ABVS.CreateVote(V, vote_value, vit, params) - > v*, which is a procedure that for the given voting *V* creates and returns a vote *v* with *vote_value* and *vit* code.

3 *ABVS.SentVote(v, known_nodes)*, which is a procedure that sends the vote *v* to the set of known node for verification and mining.

4.4 Model of auditable blockchain voting initial system testing

Before a complete prototype of ABVS was completed, some initial tests were conducted in order to identify a reference point for equipment needed for an implementation and real voting. Two main characteristics were tested:

1 blockchain validation time

2 RAM space required by the ABVS blockchain.

For the testing purposes, the following machine was used:

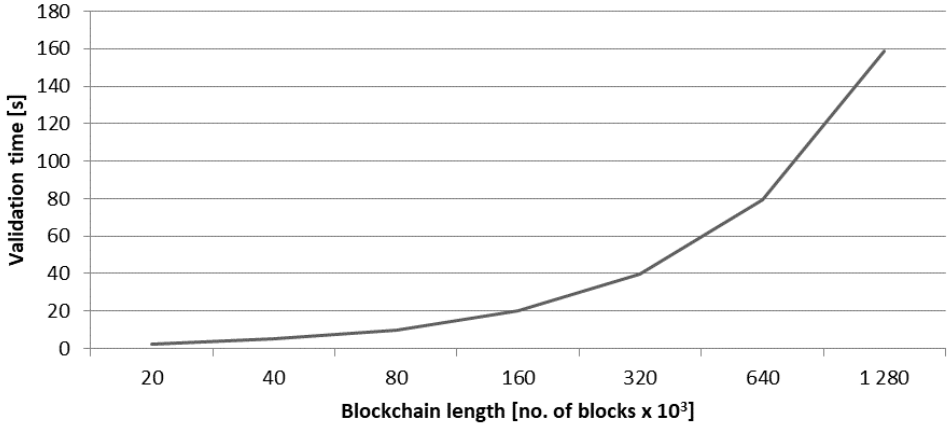
1 Processor: Intel(R) Core(TM) i5-7300 CPU @ 2.60 GHz 2.70 GHz

2 RAM: 32.0 GB (31.8 GB usable)

3 System type: 64-bit operating system, x64-based processor.

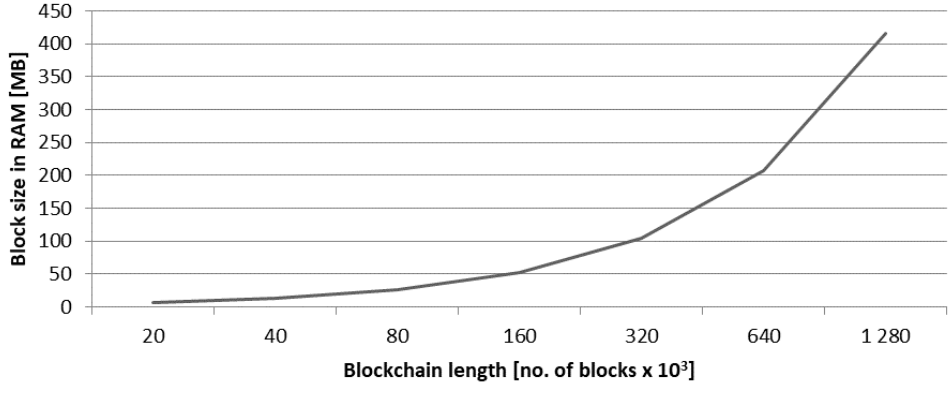
The tests were conducted on a prototype implemented in Python programming language version 3.6. Chains with block number of 20,000, 40,000, 80,000, 160,000, 320,000, 640,000 and 1,280,000 were examined. Chains with length greater than 1,280,000 blocks could not be tested due to a memory error caused by longer chains. Figure 7 illustrates averages obtained form 40 separate testing cycles. The presented values provide information on how much time would be required to obtain voting results from a single ABVS node. It is relatively quick process that took 2.46 and 158.51 seconds for the shortest and longest chains, respectively.

Figure 7 Time of validation with respect to blockchain length



In Figure 8, a diagram of blockchain size with respect to blockchain length is presented. The shortest one takes 6.5 MB of RAM memory and the longest requires 415.1 MB of RAM memory. It is important to note that processing chains longer than 1,280,000 blocks result in memory errors on the testing machine. For this reasons, tests of longer chains were not conducted. The memory errors may be the result of limitations enforced on a single process by an operating system or of an optimisation problem.

Figure 8 Blockchain RAM size with respect to blockchain length



5 Advantages and disadvantages of ABVS

ABVS is end-to-end verifiable due to usage of blockchain technology backed with VVPATs and VITs. Combination of these solutions and technologies allows the voters to follow and control their votes during and after any election. ABVS does not depend on any dedicated hardware and can be deployed both on standard computers and dedicated voting machines. The only requirement is connection to the internet. The described system is also protected against fake votes being added to the pool due multiple ways, in which number and authenticity of the votes is checked. Finally, the system is well protected against unauthorised modification because of inherent properties of blockchain technology and VVPATs utilisation as an integral part of the ABVS voting procedure.

The ABVS aimed to improve the existing voting procedure without changing it too drastically from the traditional approach. This makes adjusting to the new approach easier and decreases time required to learn it. This simplicity and gradual change is not always the case with the existing solutions, for instance, FollowMyVote.

Most importantly, majority of the systems presented in Section 3 leave identification and authentication of the voters to the election officials or force the voters to identify in person. Although it is a very effective method, it removes any advantages of remote voting because the voters must complete a non-remote identification and authentication procedures. Such systems are no different from the traditional supervised and non-remote voting approaches or their modified version proposed by ABVS. A very important advantage of ABVS is prevention of creation of fake votes cast by impersonated users, which is a real threat in the systems using remote authentication methods (e.g., ID cards or digital signatures). Voter impersonation could be accomplished by any entity with access to personal information of the voters, for instance, the government. In the recent years, election turnout in Poland ranges between 40% and 50%. This leaves almost half of the eligible voters available for using in creation of fake votes.

However, ABVS must make some trade-offs to achieve its requirements. The first one is lack of possibility of non-supervised voting because identification and authentication processes require presence of the election officials. Furthermore, voter privacy assurance is violated because the voters can be connected to their votes via VITs. This opens a possibility of coercion. Finally, ABVS needs additional infrastructure and technical staff to properly operate. This in turn created additional costs due to equipment and training needs.

6 Conclusions

Four main methods of e-voting exist: voting with dedicated voting machines, voting with optical scanning voting machines, voting with electronic ballot printers and voting through the internet. Each approach results in different benefits but also issues, which are not always easy to solve. Moreover, the e-voting field is very fragmented and each implementation is different. This is a result of lack of standards and widely accepted norms.

Most of the existing today voting systems suffer from inadequate transparency and lack of audit capabilities. Despite being the most important democratic process, voting is outside the control of common voters. They are not able to inspect and verify, if the voting process was conducted correctly or whether their votes were really included in a

vote pool. The common voters have to rely on election officials' honesty, which is often not enough to build trust in a democratic system.

Blockchain technology is a potential solution to these issues. The technology can be integrated into e-voting which in turn may provide the voters with audit capabilities and ability to supervise their votes. Blockchain-based e-voting system would reduce risk of election frauds and manipulations.

Presented in this paper ABVS is an attempt to integrate blockchain technology into an existing voting process. ABVS is a remote and supervised internet voting system that is an end-to-end auditable. It allows the common voters to follow their votes and verify the election results themselves. It is achieved by providing a public access to the blockchain storing votes after the election, VITs (unique alphanumerical codes) for vote identification and voter-verified paper audit trails for blockchain verification in case of errors.

This paper presented an overview of the whole ABVS e-voting system. The described system fulfils all of the e-voting system requirements and fulfils its goal of providing full audit and verification capabilities. However, ABVS makes trade-off when it comes to a voter privacy which is violated due to the connection between the voters, their votes and their VITs. In the future works, some alternative solution to this problem should be researched. Finally, some initial tests were conducted which resulted in identification of an equipment performance reference point which will facilitate further research.

References

Ad Hoc Committee of Experts on Legal, Operational and Technical Standards for E-voting (CAHVE) (2017) *Explanatory Memorandum to Recommendation CM/Rec(2017)5 of the Committee of Ministers to Member States on Standards for E-voting*, 14 June [online] https://search.coe.int/cm/Pages/result_details.aspx?ObjectID=090000168071bc84 (accessed 26 January 2018).

Agora Technologies (2015) *Agora_Whitepaper_v0.2.pdf* [online] https://agora.vote/Agora_Whitepaper_v0.2.pdf (accessed 20 April 2018).

Akwei, I. (2018) *Sierra Leone is First Country in the World to Use Blockchain Technology to Vote*, 15 March [online] https://face2faceafrica.com/article/sierra-leone-first-country-world-use-blockchain-technology-vote (accessed 22 April).

Bistarelli, S., Mantilacci, M., Santancini, P. and Santini, (2017) 'An end-to-end voting-system based on Bitcoin', *in Proceedings of the Symposium on Applied Computing – SAC '17*, New York, New York, USA.

Caarls, S. (2010) *E-voting Handbook: Key Steps in the Implementation of E-enabled Elections*, November, Council of Europe [online] https://www.coe.int/t/dgap/goodgovernance/Activities/E-voting/E-voting%202010/Biennial_Nov_meeting/ID10322%20GBR%206948%20Evoting%20handbook%20A5%20HD.pdf (accessed January 2018).

Castro, M. and Liskov, B. (1999) 'Practical byzantine fault tolerance', *in Proceedings of the Third Symposium on Operating Systems Design and Implementation*, New Orleans, USA.

Coin Sciences (2015) *MultiChain*, Coin Sciences [online] https://www.multichain.com/ (accessed 20 April 2018).

Council of Europe – Committee of Ministers (2017) *Recommendation CM/Rec(2017)51 of the Committee of Ministers to Member States on Standards for E-voting*, 14 June [online] https://search.coe.int/cm/Pages/result_details.aspx?ObjectID=0900001680726f6f (accessed 26 January 2018).

De Faveri, C., Moreira, A. and Araújo, J. (2016) 'Towards security modeling of e-voting systems', *in 2016 IEEE 24th International Requirements Engineering Conference Workshops (REW)*, Beijing, China.

Drescher, D. (2017) *Blockchain Basics: A Non-technical Introduction in 25 Steps*, 1st ed., Apress, Frankfurt am Main.

Enterprise Estonia (2012) *Factsheet on Estonian Blockchain Technology*, in English [online] https://e-estonia.com/wp-content/uploads/facts-a4-v03-blockchain.pdf (accessed 8 February 2018).

Ethereum Foundation (2014) *Ethereum Project*, August, Ethereum Foundation [online] https://www.ethereum.org/ (accessed 20 April 2018).

Follow My Vote, *The Online Voting Platform of the Future – Follow My Vote*, Follow My Vote [online] https://followmyvote.com (accessed 26 January 2018).

Fouard, L., Duclos, M. and Lafourcade, P. (2017) *Survey on Electronic Voting Schemes*.

Heiberg, S. and Willemson, J. (2014) 'Verifiable internet voting in Estonia', *in 2014 6th International Conference on Electronic Voting: Verifying the Vote (EVOTE)*, Lochau, Austria.

Karame, G. and Audroulaki, E. (2016) *Bitcoin and Blockchain Security*, Artech House, Inc., Norwood, MA.

Lehoucq, F. (2003) 'Electoral fraud: causes, types, and consequences', *Annual Review of Political Science*, June, Vol. 6, No. 1, pp.233–256.

Mello-Stark, S. and Lamagna, E.A. (2017) 'The need for audit-capable e-voting systems', *in 2017 31st International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, Taipei, Taiwan.

Mingxiao, D., Xiaofeng, M., Zhe, Z. and Qijun, C. (2017) 'A review on consensus algorithm of blockchain', *in 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Banff, AB, Canada.

Morabito, V. (2017) 'The security of blockchain systems', in *Business Innovation Through Blockchain*, pp.61–78, Springer, Cham.

Nakamoto, S. (2008) *Bitcoin: A Peer-to-peer Electronic Cash System* [online] https://bitcoin.org/bitcoin.pdf (accessed 20 April 2018).

National Democratic Institute, *Common Electronic Voting and Counting Technologies* [online] https://www.ndi.org/e-voting-guide/common-electronic-voting-and-counting-technologies (accessed 22 January 2018).

Ochoa, X. and Peláez, E. (2017) 'Affordable and secure electronic voting for university elections: the SAVE case study', *in 2017 Fourth International Conference on eDemocracy & eGovernment (ICEDEG)*, Quito, Ecuador.

Ojo, A. and Adebayo, S. (2017) 'Blockchain as a next generation government information infrastructure: a review of initiatives in D5 countries', in *Government 3.0 – Next Generation Government Technology Infrastructure and Services*, pp.283–298, Springer, Cham.

Pawlak, M., Guziur, J. and Poniszewska-Marańda, A. (2018a) 'Voting process with blockchain technology: auditable blockchain voting system', *Advances in Intelligent Networking and Collaborative Systems. INCoS 2018. Lecture Notes on Data Engineering and Communications Technologies*, Vol. 23, pp.233–244.

Pawlak, M., Guziur, J. and Poniszewska-Marańda, A. (2018b) 'Towards the blockchain technology for system voting process', *Cyberspace Safety and Security. CSS 2018. Lecture Notes in Computer Science*, pp.209–223.

Risius, M. and Spohrer, K. (2017) 'A blockchain research framework – what we (don't) know, where we go from here, and how we will get there', *Business & Information Systems Engineering*, December, Vol. 59, No. 6, pp.385–409.

Schneider, A., Meter, C. and Hagemeister, P. (2017) *Survey on Remote Electronic Voting*, arXiv preprint arXiv:1702.02798.

Stallings, W. (2013) 'Cryptographic hash functions', in *Cryptography and Network Security: Principles and Practice*, 6th ed., Chapter 11, pp.313–354, Pearson Education, Inc.

State Electoral Office of Estonia (2017) *General Framework of Electronic Voting and Implementation Thereof at National Elections in Estonia*, 20 June [online] https://www.valimised.ee/sites/default/files/uploads/eng/IVXV-UK-1.0-eng.pdf (accessed 22 April 2018).

United Nations Development Program (2016) *Feasibility Study on Internet Voting for the Central Electoral Commission of the Republic of Moldova: Report and Preliminary Roadmap*, Central Electoral Commission of the Republic of Moldova, Chisinau.

Willemson, J. (2018) 'Bits or paper: which should get to carry your vote?', *Journal of Information Security and Applications*, February, Vol. 38, pp.124–131.

Wolf, P., Nackerdien, R. and Tuccinardi, D. (2011) *Introducing Electronic Voting: Essential Considerations*, 1 December, International Institute for Democracy and Electoral Assistance [online] https://www.idea.int/publications/catalogue/introducing-electronic-voting-essential-considerations (accessed 22 January 2018).

Xu, X., Weber, I., Staples, M., Zhu, L., Bosch, J., Bass, L., Pautasso, C. and Rimba, P. (2017) 'A taxonomy of blockchain-based systems for architecture design', *in 2017 IEEE International Conference on Software Architecture (ICSA)*, Gothenburg, Sweden.

Zhao, J.L., Fan, S. and Yan, J. (2016) 'Overview of business innovations and research opportunities in blockchain and introduction to the special issue', in *Financial Innovation*, pp.2–28, Springer, Berlin, Heidelberg.

Zheng, Z., Xie, S., Dai, H., Chen, X. and Wang, H. (2017) 'An overview of blockchain technology: architecture, consensus, and future trends', *in 2017 IEEE International Congress on Big Data (BigData Congress)*, Honolulu, HI, USA.

Zhou, Y., Zhou, Y., Chen, S. and Wu, S.S. (2016) 'MVP: an efficient anonymous e-voting protocol', *in 2016 Global Communications Conference (GLOBECOM)*, Washington, DC, USA.